

CLAIMS

1. A process for intelligent server streaming of conventionally coded application programs across a computer network while concurrently executing said application programs on a client in a computer environment, comprising the steps of:

installing a portion of an application program on said client;

providing an application server;

partitioning said application program into appropriate page segments on said application server;

wherein said application server streams said page segments to said client upon said client's request;

executing said application program on said client;

wherein the user starts said application program in the same manner as if said application program were fully installed on said client;

wherein specific page segments are requested by said client's file system during execution of said application program; and

storing said page segments in a cache on said client.

2. The process of claim 1, wherein said application program is not recompiled, rewritten, or rebuilt for this specific delivery mechanism.

3. The process of claim 1, wherein said client manages said cache by purging page segments that are stale or not needed.

4. The process of claim 1, wherein said client does not request page segments of said application program that already reside in said cache.

5. The process of claim 1, further comprising the step of:

providing a subscription server; and

wherein the user subscribes or unsubscribes to application programs with said subscription server.

6. The process of claim 1, further comprising the step of:
providing a license server; and

wherein said client obtains an access token for a requested application program from said license server if the user has a valid subscription to said requested application program.

5 7. The process of claim 6, wherein said access token contains an expiration tag.

8. The process of claim 6, wherein said access token is securely encrypted .

10 9. The process of claim 6, wherein said client passes said access token to said application server before requesting page segments of said application program.

15 10. The process of claim 6, wherein if said license server fails said client automatically switches to another license server.

20 11. The process of claim 1, further comprising the step of:
providing a profile information database characterizing the typical page segment needs of each application program on said application server.

25 12. The process of claim 11, wherein said profile information database is updated dynamically as page segments are requested from said application server.

30 13. The process of claim 11, wherein said client prefetches page segments of said application program from said application server based on the profile information of said application program.

35 14. The process of claim 11, wherein said application server pushes page segments of said application program to said client based on the profile information of said application program.

15. The process of claim 1, wherein said client performs load balancing among a plurality of application servers for page segment requests.

16. An apparatus for intelligent server streaming of conventionally coded application programs across a computer network while concurrently executing said application programs on a client in a computer environment, comprising:
a module for installing a portion of an application program on said client;

an application server;
partitioning said application program into appropriate page segments on
said application server;
wherein said application server streams said page segments to said client
upon said client's request;
a module for executing said application program on said client;
wherein the user starts said application program in the same manner as if
said application program were fully installed on said client;
wherein specific page segments are requested by said client's file system
during execution of said application program; and
a module for storing said page segments in a cache on said client.

17. The apparatus of claim 16, wherein said application program is not
recompiled, rewritten, or rebuilt for this specific delivery mechanism.

18. The apparatus of claim 16, wherein said client manages said cache by
purging page segments that are stale or not needed.

19. The apparatus of claim 16, wherein said client does not request page
segments of said application program that already reside in said cache.

20. The apparatus of claim 16, further comprising:
a subscription server; and
wherein the user subscribes or unsubscribes to application programs with
said subscription server.

21. The apparatus of claim 16, further comprising:
a license server; and
wherein said client obtains an access token for a requested application
program from said license server if the user has a valid subscription to said
requested application program.

22. The apparatus of claim 21, wherein said access token contains an
expiration tag.

23. The apparatus of claim 21, wherein said access token is securely
encrypted.

24. The apparatus of claim 21, wherein said client passes said access token to said application server before requesting page segments of said application program.

25. The apparatus of claim 21, wherein if said license server fails said client automatically switches to another license server.

26. The apparatus of claim 16, further comprising:

a profile information database characterizing the typical page segment needs of each application program on said application server.

27. The apparatus of claim 26, wherein said profile information database is updated dynamically as page segments are requested from said application server.

28. The apparatus of claim 26, wherein said client prefetches page segments of said application program from said application server based on the profile information of said application program.

29. The apparatus of claim 26, wherein said application server pushes page segments of said application program to said client based on the profile information of said application program.

30. The apparatus of claim 16, wherein said client performs load balancing among a plurality of application servers for page segment requests.

31. A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform method steps for intelligent server streaming of conventionally coded application programs across a computer network while concurrently executing said application programs on a client in a computer environment, comprising the steps of:

installing a portion of an application program on said client;

providing an application server;

partitioning said application program into appropriate page segments on said application server;

wherein said application server streams said page segments to said client upon said client's request;

executing said application program on said client;

wherein the user starts said application program in the same manner as if said application program were fully installed on said client;

wherein specific page segments are requested by said client's file system during execution of said application program; and

5 storing said page segments in a cache on said client.

32. The method of claim 31, wherein said application program is not recompiled, rewritten, or rebuilt for this specific delivery mechanism.

10 33. The method of claim 31, wherein said client manages said cache by purging page segments that are stale or not needed.

34. The method of claim 31, wherein said client does not request page segments of said application program that already reside in said cache.

15 35. The method of claim 31, further comprising the step of:
providing a subscription server; and
wherein the user subscribes or unsubscribes to application programs with said subscription server.

20 36. The method of claim 31, further comprising the step of:
providing a license server; and
wherein said client obtains an access token for a requested application program from said license server if the user has a valid subscription to said requested application program.

25 37. The method of claim 36, wherein said access token contains an expiration tag.

30 38. The method of claim 36, wherein said access token is securely encrypted .

39. The method of claim 36, wherein said client passes said access token to said application server before requesting page segments of said application program.

35 40. The method of claim 36, wherein if said license server fails said client automatically switches to another license server.

41. The method of claim 31, further comprising the step of:

providing a profile information database characterizing the typical page segment needs of each application program on said application server.

42. The method of claim 41, wherein said profile information database is updated dynamically as page segments are requested from said application server.

43. The method of claim 41, wherein said client prefetches page segments of said application program from said application server based on the profile information of said application program.

44. The method of claim 41, wherein said application server pushes page segments of said application program to said client based on the profile information of said application program.

45. The method of claim 31, wherein said client performs load balancing among a plurality of application servers for page segment requests.